Algoritmos no App Inventor

APRESENTAÇÃO

O uso de aplicativos aumentou consideravelmente nos últimos tempos. Consequentemente, a demanda pelo desenvolvimento desses artefatos tecnológicos acabou desafiando a capacidade de produção de *softwares*. Para isso, muitas empresas optam pela adoção de metodologias ágeis, as quais garantirão uma quantidade maior de *softwares* produzidos em pouco tempo e sem perda de qualidade.

Nesta Unidade de Aprendizagem, você irá aprender os principais conceitos do MIT App Inventor, o qual foi desenvolvido pelo Google com o intuito de proporcionar aos desenvolvedores um ambiente lúdico para a criação de aplicativos para dispositivos Android. Para isso, você irá conhecer sua interface, a qual está inserida em um ambiente *web*, e alguns exemplos práticos.

Bons estudos.

Ao final desta Unidade de Aprendizagem, você deve apresentar os seguintes aprendizados:

- Definir os aspectos básicos do App Inventor.
- Descrever a interface do App Inventor (App Inventor Designer e Blocks Editor).
- Praticar a utilização dos recursos do App Inventor (exemplo prático com o emulador).

INFOGRÁFICO

Você pode perceber que a evolução é necessária quando a demanda por novidades ressalta a necessidade de novos recursos. Com o MIT App Inventor não é diferente.

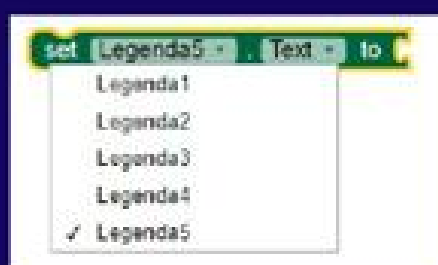
Confira, no Infográfico, os novos recursos presentes no App Inventor 2, como blocos, variáveis, cores, dentre outros.

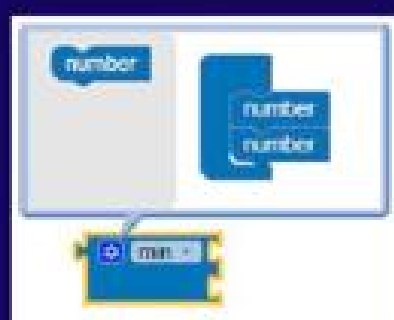
RECURSOS DO APP INVENTOR 2

Assim como as demais ferramentas de desenvolvimento, o App Inventor passou por algumas atualizações, tanto que atualmente se encontra em sua segunda versão.

Confira:

DROPDOWNS

São blocos podem ser alterados para significar ou fazer outra coisa apenas clicando no nome do bloco.

MUTATORS

Permitem que determinados blocos se expandam, diminuam ou até mesmo alterem sua funcionalidade.

GLOBAL VS LOCAL VARIABLES (VARIÁVEIS GLOBAIS E LOCAIS)

Uma variável global é aquela que pode ser acessada em qualquer local do programa.

Já uma variável local só é acessada dentro da função em que foi inserida.

COLORS

BLOCO BÁSICO DE CORES

Há uma forma quadrada pequena e há uma cor no meio, que representa a cor armazenada internamente neste bloco. Ao clicarmos na cor no meio, um *pop-up* aparece na tela com uma tabela de 70 cores disponíveis para que possamos escolher.

PODEMOS TAMBÉM FAZER CORES

Para isso, podemos inserir as numerações correspondentes às cores. Esses números da lista representam valores em um **código RGB (abreviação para um sistema de cores aditivas, R (*red*), G (*green*), B (*blue*).**

make color | make a list | 255 → R | 0 → G | 0 → B

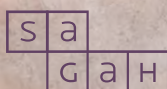
CONTEÚDO DO LIVRO

A possibilidade de termos autonomia é sempre genial. Agora imagine se essa autonomia está atrelada ao desenvolvimento de um aplicativo, que pode ser criado para sanar uma necessidade particular sua, e ser feito do seu jeito. O App Inventor traz esta ideia, já que ele proporciona que possamos desenvolver aplicativos por meio lúdico e em um ambiente diferenciado.

No capítulo Algoritmos no App Inventor, da obra *Pensamento computacional*, você verá os seus conceitos e recursos do App Inventor em exemplo prático.

Boa leitura.

PENSAMENTO COMPUTACIONAL

Izabelly Soares de Moraes

Algoritmos no App Inventor

Objetivos de aprendizagem

Ao final deste texto, você deve apresentar os seguintes aprendizados:

- Definir os aspectos básicos do App Inventor.
- Descrever a interface do App Inventor (*App Inventor Designer* e *Blocks Editor*).
- Praticar a utilização dos recursos do App Inventor (exemplo prático com o emulador).

Introdução

Atualmente, a presença da tecnologia é primordial para que possamos realizar a maioria de nossas atividades diárias. O que não imaginamos, na maior parte das vezes, é que esses artefatos tecnológicos dos quais fazemos uso demandam o uso de linguagens de programação e ferramentas capazes de originar todas as funcionalidades responsáveis por atender à nossa demanda.

Neste capítulo, você vai aprender sobre o MIT App Inventor, um ambiente de programação que propicia a criação de aplicativos por meio de uma interface acessível e com o objetivo de proporcionar a todos a possibilidade de desenvolver um aplicativo para o sistema operacional Android. Para isso, você vai ver, além de seus principais conceitos, sua interface e exemplos práticos.

Aspectos básicos do App Inventor

Com o passar dos anos, a sociedade moderna — hoje, uma "sociedade tecnológica" — começou, cada vez mais, a demonstrar a necessidade de artefatos que suprissem as necessidades diárias de uma forma eficiente e eficaz. Foi no sentido de contribuir para essa demanda que o MIT App Inventor, foco deste capítulo, foi criado. No entanto, antes de nos aprofundarmos sobre o contexto do MIT App Inventor em si, é importante considerarmos outros aspectos,

como o contexto que envolve o uso desse ambiente, que é o da programação de computadores. Se observarmos as nossas atividades cotidianas, veremos que estamos sempre pedindo auxílio a um recurso computacional, de modo que a solução de nossas necessidades se dá por meio das funcionalidades existentes nesses artefatos. Assim, a programação nos proporciona o desenvolvimento de diversas coisas por meio do uso das linguagens de programação.

Hoje, um dos maiores desafios das empresas dedicadas ao desenvolvimento de *softwares* é, justamente, o de atender à alta demanda. A nossa necessidade diante de um *software* pode ser exemplificada pelo próprio navegador da internet, o que chamamos de *browser.* Antigamente, ele era utilizado apenas para acessarmos os sites, mas, hoje, é capaz de armazenar nossos dados, criar listas de favoritos, permite-nos fazer login em nossos e-mails e salva todas as nossas informações (caso necessitemos atualizar ou mudar de *browser*), dentre outras funcionalidades. Se você observar, verá que esse acréscimo de possibilidades de uso ocorreu praticamente em todos os aplicativos que utilizamos ou fomos utilizando ao longo de nossas experiências tecnológicas.

Diante da notória dificuldade que pessoas comuns, ou seja, que não têm formação ou dedicação a assuntos relacionados ao desenvolvimento de *softwares*, demonstram ter em compreender a lógica da programação, diversos grupos de pesquisa de instituições de ensino e até mesmo grandes empresas têm empenhado-se em facilitar essa atividade.

Em 2010, o aplicativo antes denominado apenas App Inventor, foi disponibilizado por Hal Abelson e Mark Friedman. Posteriormente, a Google realizou o lançamento do código-fonte e financiou o desenvolvimento e a evolução dessa ferramenta. Hoje, ela é tida como foco de estudos de diversas instituições de ensino por todo o mundo e é mantida pelo Massachusetts Institute of Technology (MIT).

O MIT App Inventor é um ambiente de programação visual e intuitivo que proporciona o desenvolvimento de aplicativos totalmente funcionais para *smartphones* e *tablets*. O objetivo, como citado anteriormente, é democratizar o processo de desenvolvimento de *softwares* e incentivar a autonomia de todos, na criação de aplicativos, nesse caso, para o sistema operacional Android. Assim, ele utiliza artefatos gráficos para representar as instruções de programação e traz o conceito de "arrastar e soltar" as peças, como se fosse um quebra-cabeça.

Fique atento

O ambiente de desenvolvimento do App Inventor é on-line, ou seja, basta acessar o site **https://goo.gl/5sdu1j** por meio de algum dos seguintes *browsers*: Mozilla Firefox, Google Chrome ou Apple Safari. A partir disso, é possível inicializar a criação dos aplicativos.

É importante saber que você pode acessá-lo utilizando algum dos seguintes sistemas operacionais: Linux (Ubuntu 8 ou superior, Debian 5 ou superior), Windows (a partir do XP) e PC (Macintosh com processador Intel), Mac OS X 10.5 ou superior. A faixa etária indicada é a partir dos 9 anos de idade.

Agora que você já conhece um pouco sobre a principal definição e funcionalidade do App Inventor, verá, na próxima seção, sua interface gráfica e seus recursos. Não podemos esquecer que, assim como outros recursos tecnológicos, o App Inventor sempre está em processo de evolução, de modo que trataremos, aqui, de aspectos direcionados ao App Inventor 2, que é a versão mais atual, lançada em 2013.

Interface do App Inventor

Diferentemente dos outros ambientes de desenvolvimento de software, o App Inventor se diferencia por ser on-line, ou seja, você não precisa baixar em seu computador nenhum *software* para desenvolvimento. Dessa forma, poderá acessar seu projeto de onde e quando quiser. Além disso, por ser um ambiente mantido pelo MIT, ainda temos a possibilidade de compartilhar nossos projetos com outros desenvolvedores. O objetivo é trazer o clima de desenvolvimento inspirador e colaborativo.

Mas você sabe como pode ter acesso ao App Inventor? A resposta é simples: basta acessar, por meio do *browser*, o seguinte link: **https://goo.gl/2bnIh**. Após esse acesso, você verá a página inicial do App Inventor, como mostra a Figura 1.

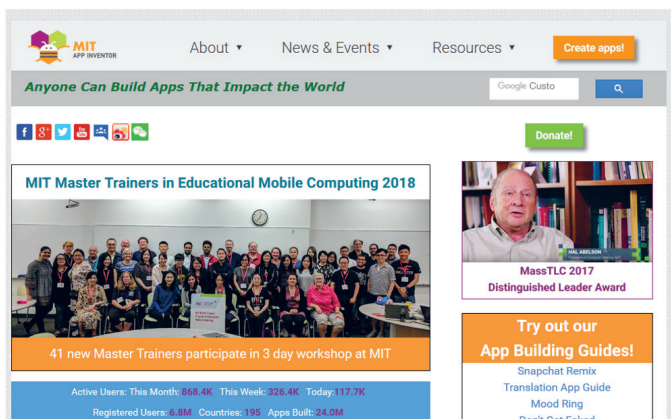



Figura 1. Página inicial do App Inventor.

Fonte: MIT App Inventor (2018, documento on-line).

Saiba mais

Na página inicial, você também pode ter acesso, dentre outras informações, a tutoriais, fóruns e instruções para que possa criar os seus primeiros aplicativos.

No canto superior direito dessa primeira página, podemos notar a opção “Create Apps” (criar aplicativos), como mostra a Figura 2.

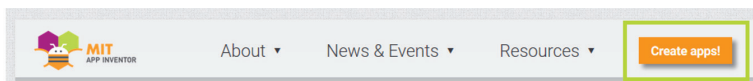



Figura 2. Menu com as opções da página inicial.

Fonte: MIT App Inventor (2018, documento on-line).

Ao selecionar essa opção, você será direcionado a uma página para realizar a inserção de seu e-mail e de sua senha. Já que você utilizará um ambiente web para desenvolver os aplicativos, é importante realizar esse login, de modo

que os projetos desenvolvidos fiquem associados a você. Um próximo passo é realizar a leitura dos Termos de Serviço e, por fim, aceitá-lo (Figura 3). Posteriormente, você deve familiarizar-se com a página na qual terá a possibilidade de gerenciar os projetos, tanto para criar novos quanto para gerenciar os que já desenvolveu. Outra possibilidade é a de escolher o idioma, já que, inicialmente, a página apresenta o idioma inglês. Nesse mesmo espaço, você também tem a opção de publicar os projetos em uma galeria. Para facilitar o seu entendimento, vamos mudar o idioma para o português?

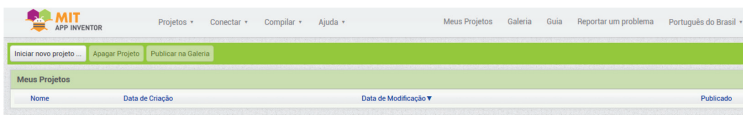



Figura 3. Página inicial, após o aceite do Termo de Serviço.

Fonte: MIT App Inventor (2018, documento on-line).

Fique atento

Caso você já possua algum projeto criado anteriormente, poderá acessar a opção de "Projetos" e, posteriormente, clicar na opção "Meus projetos". É nessa mesma opção de menu que você poderá importar seu projeto, apagá-lo, dentre outras opções.

Ao selecionar a opção "Iniciar novo projeto", você deverá informar qual nome será dado ao seu primeiro (ou novo) projeto (Figura 4).

Figura 4. Campo para inserção do nome do projeto.

Fonte: MIT App Inventor (2018, documento on-line).

Após inserir o nome de seu projeto, você será direcionado à página denominada App Inventor Design. Observe que, na barra de menu verde, do lado direito, você terá duas opções, como mostra a Figura 5: “Designer” e “Blocos”.

Figura 5. Interface de desenvolvimento.

Fonte: MIT App Inventor (2018, documento on-line).

Designer

A página recebe essa denominação, pois, como você pode observar na Figura 6, traz os principais recursos disponibilizados pela ferramenta para que possamos projetar a interface do aplicativo. Você deve lembrar que esses termos estão sendo utilizados porque estamos utilizando a ferramenta na língua portuguesa (do Brasil).

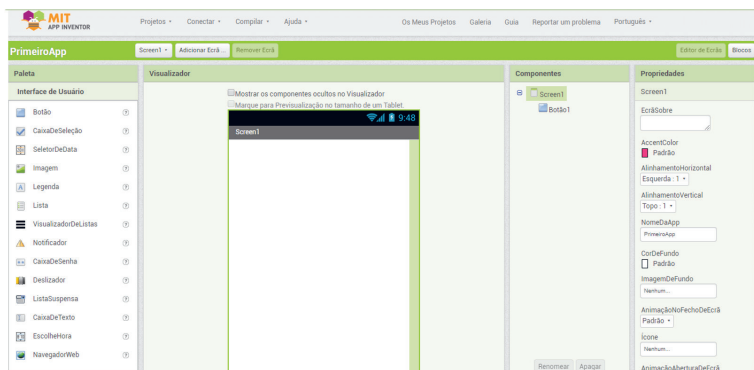



Figura 6. Interface de desenvolvimento — Designer.

Fonte: MIT App Inventor (2018, documento on-line).

A seguir, veja os recursos dessa ferramenta.

Paleta

É uma barra de menu que expõe visual e graficamente os componentes que o usuário tem à disposição para desenvolver seu aplicativo. Como componentes, podemos listar botões, campos de texto, imagens, lista, caixa de seleção, dentre outros (Figura 7).

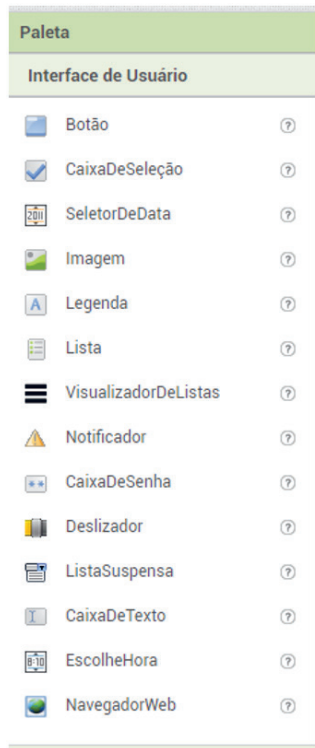



Figura 7. Barra de menu — Paleta.

Fonte: MIT App Inventor (2018, documento on-line).

Link

Para saber mais informações sobre os componentes e suas respectivas particularidades, acesse o link a seguir.

https://goo.gl/zSwAVp

Visualizador

É o espaço no qual se inserem os componentes. O App Inventor trabalha com o mesmo conceito de quebra-cabeça, em que os componentes se encaixam de forma lógica. Aqui, eles são arrastados e soltos ou encaixados onde queremos que eles fiquem, de modo que possamos formar o layout do aplicativo que está sendo desenvolvido.

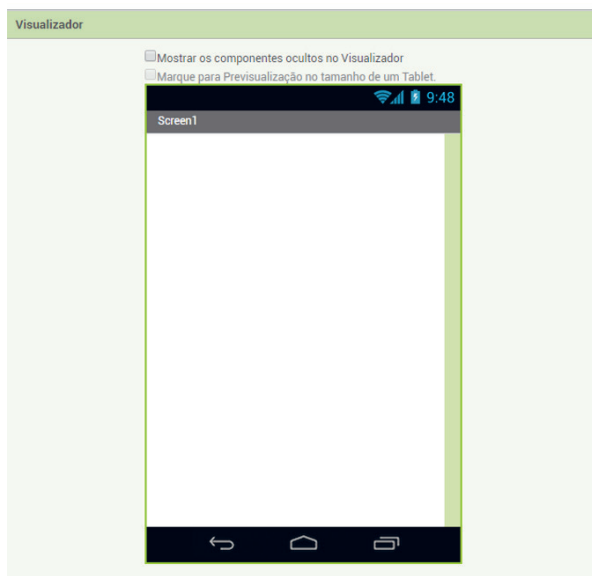



Figura 8. Área de visualização.

Fonte: MIT App Inventor (2018, documento on-line).

Componentes

É o espaço reservado para que possamos visualizar os componentes (Figura 9) que selecionamos. Nesse ambiente, você pode, também, editar os componentes e até mesmo apagá-los. Na Figura 8, você pode notar que temos a “Screen1” adicionada, ou seja, temos uma primeira tela criada. Esse espaço é importante, pois, no mesmo aplicativo, podemos ter diversas telas, conforme a quantidade de funcionalidades e como a navegabilidade será determinada pelo desenvolvedor. Dessa forma, temos como acompanhar quais componentes foram inseridos e a forma como eles estão nomeados.

Figura 9. Componentes.
Fonte: MIT App Inventor (2018, documento on-line).

Propriedades

Possibilitam a mudança das propriedades/características dos componentes, como mudança de cores, tamanhos, imagens de fundo, fonte da letra, dentre outros — já que essas características variam conforme o componente utilizado (Figura 10).

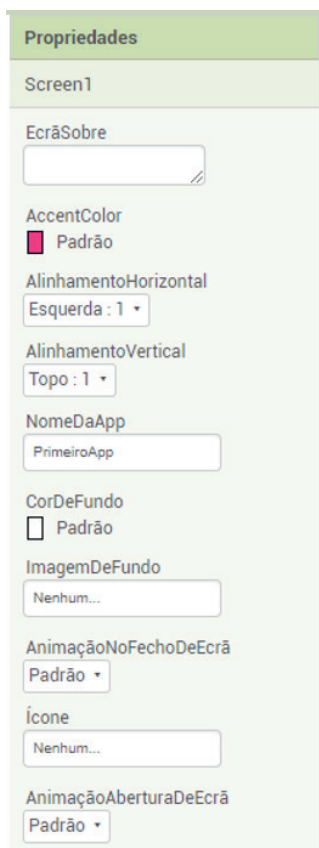



Figura 10. Barra de menu — Propriedades.
Fonte: MIT App Inventor (2018, documento on-line).

Multimédia

Nesse espaço, serão exibidos os arquivos que forem carregados para dentro do seu projeto e que podem ser de áudio, vídeo e imagens (Figura 11). Os arquivos suportados são:

- Áudio: mp3, wav, mid, 3gp, mp4, mkv, ota, imy e ogg.
- Vídeo: mp4, 3gp, webm e mkv.
- Imagens: bmp, gif, jpg, png e webp.

Figura 11. Multimédia.

Fonte: MIT App Inventor (2018, documento on-line).

Nesta seção, vimos que, ao inicializar um projeto, temos a opção de escolher qual layout queremos e citamos particularidades relacionadas ao ambiente de desenvolvimento "Designer", mais conhecido como App Inventor Design. A seguir, veremos um pouco sobre o layout de "Blocos" ou *Blocks Editor*.

Blocks Editor

Como o próprio nome já diz, o *Blocks Editor* é um espaço no qual a edição da lógica de programação ocorre por meio de blocos (Figura 12).

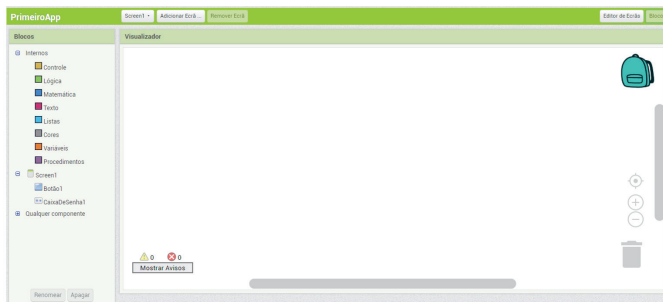



Figura 12. Tela da área dos blocos.

Fonte: MIT App Inventor (2018, documento on-line).

Nessa tela, você pode observar que temos várias possibilidades de blocos de instruções com suas configurações já predefinidas. O desenvolvedor precisa apenas selecionar o bloco que deseja utilizar e arrastá-lo para os visualizados. A seguir, na Figura 13, veja a lista de blocos disponíveis. Caso não deseje mais o bloco selecionado, você tem a opção de direcioná-lo à lixeira, seguindo a ideia da ferramenta; para que isso ocorra, é só selecionar o bloco e arrastá-lo para a lixeira. Você pode, também, centralizar, maximizar ou diminuir a exibição dos blocos.

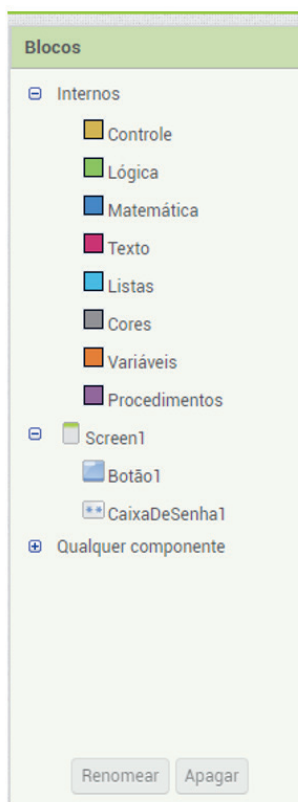



Figura 13. Blocos.

Fonte: MIT App Inventor (2018, documento on-line).

Os blocos trazem comportamentos já predefinidos pela lógica de programação. A seguir, você verá a definição de cada um deles.

Fique atento

Caso os blocos não se encaixem, é provável que seu programa esteja com algum erro. Não se esqueça que eles devem se encaixar como um quebra-cabeças, usando sempre a lógica de programação.

Controle

A estrutura de controle (Figura 14) é utilizada quando uma ou mais condições devem ser analisadas, e, a partir dessa análise, caso o resultado lógico seja verdadeiro, a sequência de execução do código poderá, ou não, sofrer alterações.

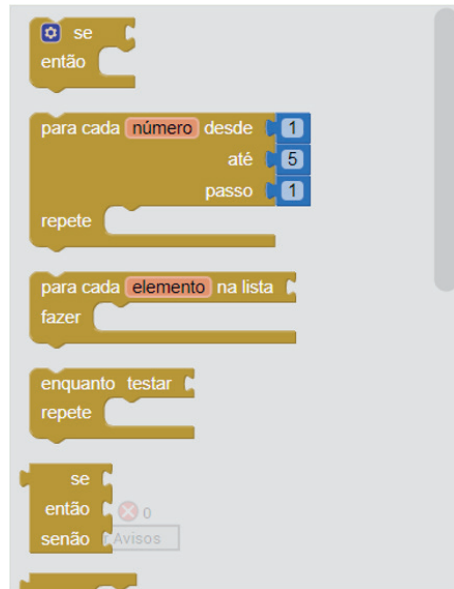



Figura 14. Blocos de controle.

Fonte: MIT App Inventor (2018, documento on-line).

A seguir, você verá algumas das estruturas de controle e, aos poucos, vai aprender as particularidades de cada uma. É importante lembrar que, por mais que o uso dessa ferramenta seja considerado mais simples do que o de muitas outras, ela traz a aplicação de conceitos da lógica de programação, que, muitas vezes, faz parte, também, da lógica matemática.

Na Figura 15, podemos observar que, ao selecionar o bloco da estrutura de controle "se... então", estaremos trabalhando com a estrutura de controle simples, em que, conforme sua sintaxe, a instrução só será executada caso a condição analisada seja verdadeira; caso não o seja, o programa terminará sua execução e os blocos de instruções não serão executados. Em inglês, essa estrutura fica com os seguintes comandos: *if... then*.

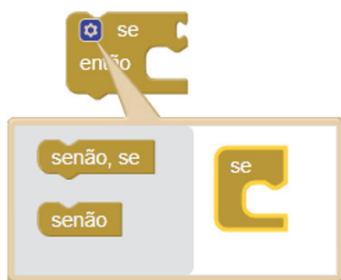



Figura 15. Bloco de controle "se... então".
Fonte: MIT App Inventor (2018, documento on-line).

A Figura 16 traz uma estrutura semelhante à apresentada na Figura 15, com a diferença de que, agora, teremos blocos de instruções a serem executados caso a condição seja verdadeira; caso a condição seja falsa, essa estrutura é chamada de estrutura de controle composta. Em inglês, essa estrutura fica com os seguintes comandos: *if... then... else*.

A complexidade da ferramenta está atrelada à robustez de recursos que ela nos disponibiliza para que possamos desenvolver nossos aplicativos. Por isso, muitos conceitos você realmente só irá aprender na prática e acessando, claro, o site da ferramenta, que apresenta a definição de todos os recursos.

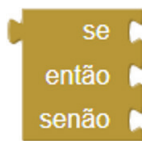



Figura 16. Bloco de controle: "se... então... senão".
Fonte: MIT App Inventor (2018, documento on-line).

Link

Mais informações sobre todos os blocos de controle e suas respectivas particularidades podem ser acessadas por meio do link a seguir.

https://goo.gl/RIlvC4

Lógica

Os blocos denominados "lógica" trazem aspectos bem ligados à lógica matemática, em que o verdadeiro e o falso remetem aos princípios de George Boole, que foi um estudioso com afinidades matemáticas e filosóficas que trouxe a ideia de que conseguiríamos expressar-nos por meio de termos matemáticos. Dessa forma, a álgebra booleana surgiu tendo como embasamento a lógica binária, ou seja, apresenta apenas duas duas possibilidades de representação.

Veja, na Figura 17, que esse quadro de blocos de lógica traz os valores "verdadeiro", "falso" (compreendem valores booleanos), "e", "ou" (são conectivos lógicos), "=". Mas o que seriam os conectivos lógicos? Eles são utilizados quando precisamos analisar mais de uma condição para a mesma tomada de decisão. Quer um exemplo prático? Quando você precisa ir a um local ao que nunca foi antes, o que você faz? Vamos supor que você utilize seu GPS. Quando colocamos um endereço e traçamos as rotas, o GPS nos fornece algumas opções e, a partir disso, escolhemos a que melhor nos convém, tendo como base algumas condições, como, por exemplo, se a estrada é asfaltada, qual é o caminho mais perto, dentre outras. Essas condições são analisadas na mesma tomada de decisão, ou seja, quando você precisa ir até seu destino final. Essa situação é bastante comum quando estamos desenvolvendo um *software*.

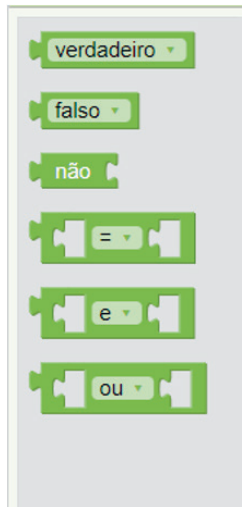



Figura 17. Blocos de lógica.

Fonte: MIT App Inventor (2018, documento on-line).

Link

Mais informações sobre todos os blocos de lógica e suas respectivas particularidades podem ser acessadas por meio do link a seguir.

https://goo.gl/wwBJ2v

Matemática

Alguns blocos matemáticos (Figura 18) são *dropdowns*, o que significa que eles podem ser convertidos em blocos diferentes.

Figura 18. Blocos da matemática.

Fonte: MIT App Inventor (2018, documento on-line).

Para isso, o desenvolvedor deve apenas selecionar a opção que ele deseja que o bloco assuma, como mostra a Figura 19.

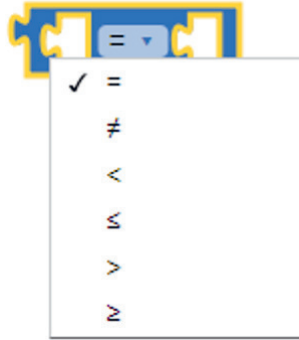



Figura 19. Bloco *dropdown* da matemática.
Fonte: MIT App Inventor (2018, documento on-line).

Link

Mais informações sobre todos os blocos da matemática e suas respectivas particularidades podem ser acessadas por meio do link a seguir.

https://goo.gl/DM7dbq

Texto

Assim como os demais blocos, o bloco de texto (Figura 20) permite que trabalhemos com o tipo de dado "*string*" (em português, caracteres). Na programação, os caracteres são compostos por valores delimitados, muitas vezes, por aspas, formados por letras, números e símbolos como @!, entre outros.

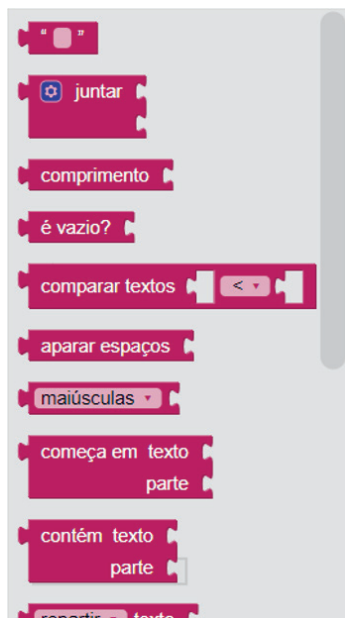



Figura 20. Blocos de texto.

Fonte: MIT App Inventor (2018, documento on-line).

Link

Acesse o link a seguir para saber mais informações sobre todos os blocos de texto e suas respectivas particularidades.

https://goo.gl/XodQqC

Listas

As listas (Figura 21) são utilizadas para que possamos armazenar e, consequentemente, manipular diferentes elementos ou tipos de dados. O conceito de lista envolve, também, a definição de um índice, ou seja, cada elemento dentro de uma lista possui um índice, que indica a posição do elemento dentro da lista.

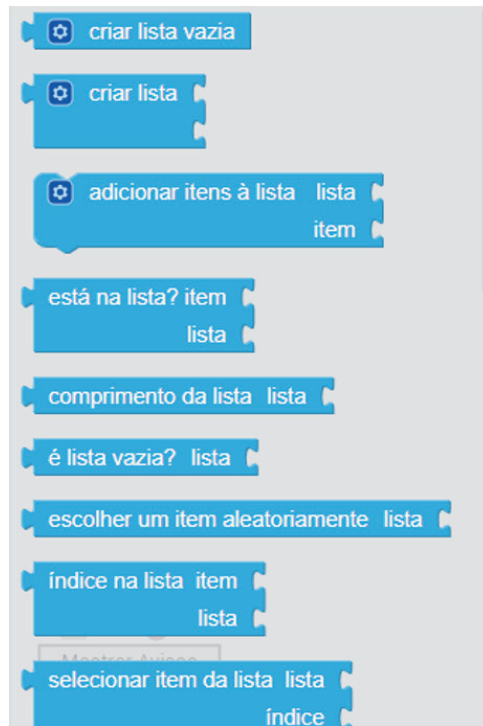



Figura 21. Bloco de listas.

Fonte: MIT App Inventor (2018, documento on-line).

Link

Mais informações sobre todos os blocos de listas e suas respectivas particularidades podem ser acessadas no link a seguir.

https://goo.gl/ZFWyPDa

Cores

Ao selecionar o bloco de cores, você terá acesso a diversas outras possibilidades, ou seja, as opções não se limitam apenas às que já são vistas quando adiciona-

mos o bloco de cores, como você pode observar na Figura 22. Internamente, o App Inventor armazena cada cor como um único número e também dá a opção de criarmos uma nova cor.

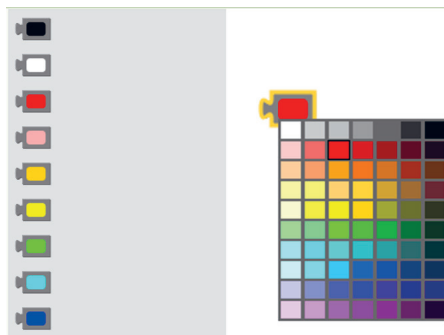



Figura 22. Bloco de cores.
Fonte: MIT App Inventor (2018, documento on-line).

Link

Para saber mais sobre todos os blocos de cores e suas respectivas particularidades, acesse o link a seguir.

https://goo.gl/uQtHHg

Variáveis

As variáveis (Figura 23) são espaços que criamos na memória do computador para que possamos armazenar dados. Falamos anteriormente de alguns tipos de dados, como os caracteres e números (inteiros e reais). São esses tipos de dados que determinam o tamanho do espaço que deve ser criado na memória do computador, ou seja, quando criamos uma variável, temos que definir qual tipo de dado ela irá receber. Essa definição afetará todo o seu uso dentro do *software* que está sendo desenvolvido.

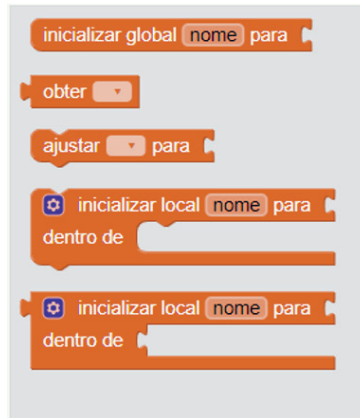



Figura 23. Bloco de variáveis.

Fonte: MIT App Inventor (2018, documento on-line).

Link

Mais informações sobre todos os blocos de variáveis e suas respectivas particularidades podem ser acessadas por meio do link a seguir.

https://goo.gl/HeJLXH

Também se recomenda que você leia os conceitos que norteiam o uso das variáveis, o que você pode fazer no seguinte link:

https://goo.gl/RtpHjG

Procedimentos

Os procedimentos (Figura 24) trazem um conjunto de instruções já predeterminadas, ou seja, você não precisa ficar sempre repetindo certas instruções se, antes, você criar um procedimento. Sempre que precisar, é só convocar o procedimento, e ele trará as instruções de que você precisa. No App Inventor, essas instruções são compostas por um conjunto de blocos. Por exemplo, você deve armazenar 100 valores; claro que temos diversas maneiras de desenvolver

essa lógica utilizando a programação, mas, para facilitar o desenvolvimento, podemos inserir as instruções de armazenamento dos valores em um procedimento e, sempre que for preciso armazenar 100 valores, chamaremos apenas o procedimento, em vez de criar uma instrução para armazenar valor por valor.

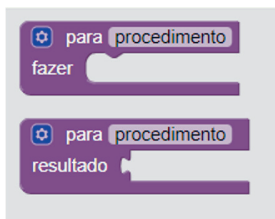



Figura 24. Bloco de procedimentos.
Fonte: MIT App Inventor (2018, documento on-line).

Link

Mais informações sobre todos os blocos de Procedures e suas respectivas particularidades podem ser acessadas por meio do link a seguir.

https://goo.gl/dW6ah3

Utilização do emulador

Vamos colocar a mão na massa? Você já viu a quantidade de recursos que a ferramenta nos disponibiliza — são muitas, não é? O segredo é compreender corretamente o que cada uma pode proporcionar, pois esse primeiro passo facilitará todo o processo de desenvolvimento do aplicativo. Ah! E não se preocupe em aprender todos esses recursos instantaneamente, já que esse processo ocorre naturalmente, aos poucos, à medida que vamos tendo contato com os recursos e suas particularidades.

Vamos criar um aplicativo básico, para que você possa familiarizar-se com a ferramenta.

Alguns passos iniciais são primordiais para que você consiga realizar os testes de seu aplicativo no emulador, que pode ser seu próprio *smarthphone*, caso ele tenha o sistema operacional Android.

1. Acesse o site **https://goo.gl/5sdu1j** e selecione a opção "Criar aplicativo", como mencionamos anteriormente.
2. Testaremos nosso aplicativo por meio de um emulador, que permite simular o uso de outro sistema. Nesse caso, é uma boa opção se você não tiver disponível algum dispositivo com o sistema operacional Android.
3. Para configurar o emulador, você deverá acessar o seguinte link: **https://goo.gl/4RdYwe**. Posteriormente, selecione o sistema operacional que você está utilizando. Observe que, no próprio site, há indicações de outras configurações necessárias para que tudo ocorra como o esperado. Utilizaremos, aqui, as configurações para o sistema operacional Windows.
4. Posteriormente, siga todos os passos indicados pelo link citado no passo 3. Ao baixar o arquivo do instalador, permita que seja realizada a sua instalação, como mostra a Figura 25.

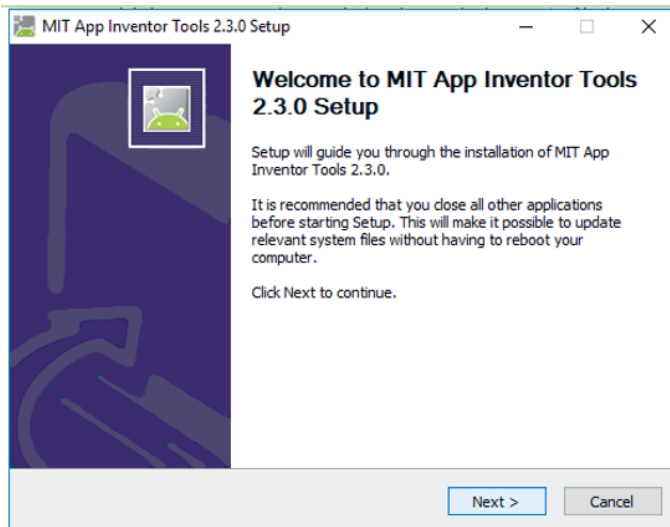



Figura 25. Instalação do emulador.

5. Ao finalizar a instalação, você terá acesso a um emulador (Figura 26).

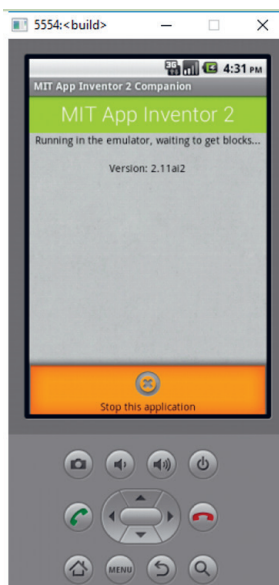



Figura 26. Emulador.

6. Agora que o emulador foi instalado em seu computador, você deverá acessar, na página web do App Inventor, no menu superior, a opção "Conectar" e, posteriormente, deverá selecionar a opção "Emulador" (Figura 27).

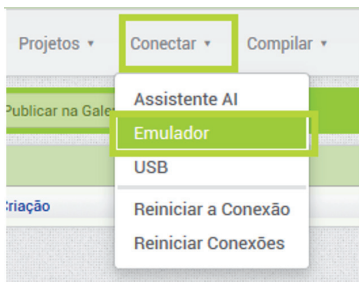
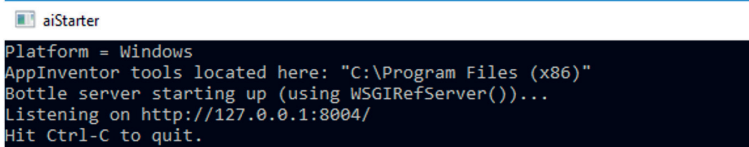



Figura 27. Seleção da opção "Conectar".

Fonte: MIT App Inventor (2018, documento on-line).

7. Ao selecionar essa opção, o sistema retornará uma mensagem pedindo que você aguarde um pouco, pois o emulador estará sendo inicializado. Caso, após a instalação, ele já esteja aberto, você pode selecionar a opção "Reiniciar conexão". Se não, é só você localizar onde o aiStarter foi instalado em seu computador. O programa aiStarter é instalado no seu computador quando você instala o *software* de instalação do App Inventor e fornece comunicação entre o App Inventor em execução no navegador e outras partes do App Inventor. Ele é bem semelhante ao *prompt* de comando, como mostra a Figura 28.

aiStarter

```
Platform = Windows
AppInventor tools located here: "C:\Program Files (x86)"
Bottle server starting up (using WSGIRefServer())...
Listening on http://127.0.0.1:8004/
Hit Ctrl-C to quit.
```

Figura 28. aiStarter.

Fonte: MIT App Inventor (2018, documento on-line).

8. Para começar a utilizar o emulador e, consequentemente, a ferramenta, vamos inserir alguns componentes.

Link

Mais instruções sobre a instalação do emulador referente a cada sistema operacional podem ser encontradas nos links a seguir:

- Sistema operacional Windows: **https://goo.gl/QSM1vM**
- Sistema operacional Mac OS X: **https://goo.gl/k5FNEp**
- Sistema operacional Linux: **https://goo.gl/g35G5g**

9. Primeiro, selecione, na Paleta, em "Interface de Usuário", um componente do tipo "Botão"; posteriormente, arraste esse componente para o Visualizador, conforme a Figura 29.

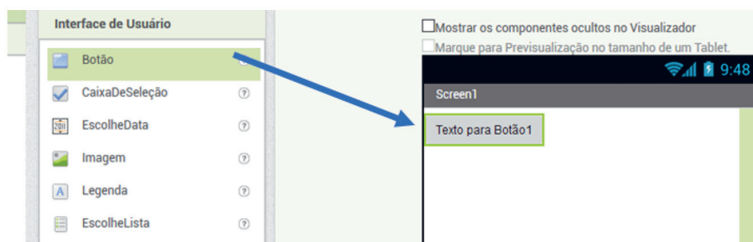



Figura 29. Uso do componente Botão.
Fonte: MIT App Inventor (2018, documento on-line).

10. Ao inserir o componente no Visualizador, perceba que podemos alterar as propriedades do componente, ao lado direito, no menu de propriedades. Lembre-se que todo componente terá suas propriedades; portanto, não esqueça de selecionar, no Visualizador, qual componente você deseja alterar. A seguir, na Figura 30, veja o exemplo da mudança da propriedade "cor" do componente Botão1.

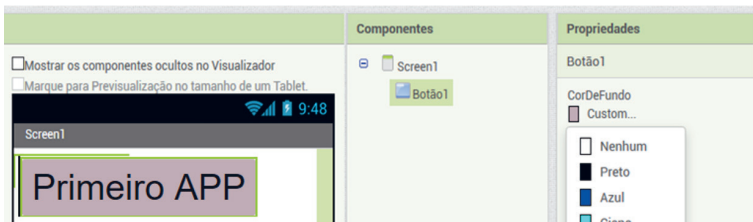



Figura 30. Propriedades do Botão1.
Fonte: MIT App Inventor (2018, documento on-line).

11. Podemos, também, mudar o texto do botão, como mostra a Figura 30, em que foi inserido o texto "Primeiro APP".

12. Posteriormente, ainda na Paleta, selecione a opção "Mídia" e, depois, "TextoparaFalar". A sua ação deve ser igual às demais; segure o componente e arraste para o Visualizador. Você notará que, na tela, não aparecerá nada, porém, na barra de componentes, você visualizará o componente associado a Screen1, ou seja, a tela 1. Não se esqueça que você pode ter informações sobre cada componente disponibilizado na

ferramenta por meio dos links que indicamos durante a leitura deste material.

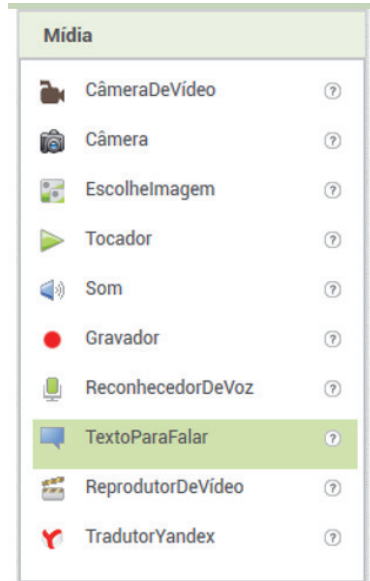



Figura 31. Paleta — Mídia.
Fonte: MIT App Inventor (2018, documento on-line).

13. O próximo passo é você seguir as indicações da Figura 32, selecionando a opção "Blocos"; no menu superior, conforme a numeração 1, e, posteriormente, selecionar o "Botão1", no menu dos "Blocos", como mostra a numeração 2.

Figura 32. Utilizando os blocos.

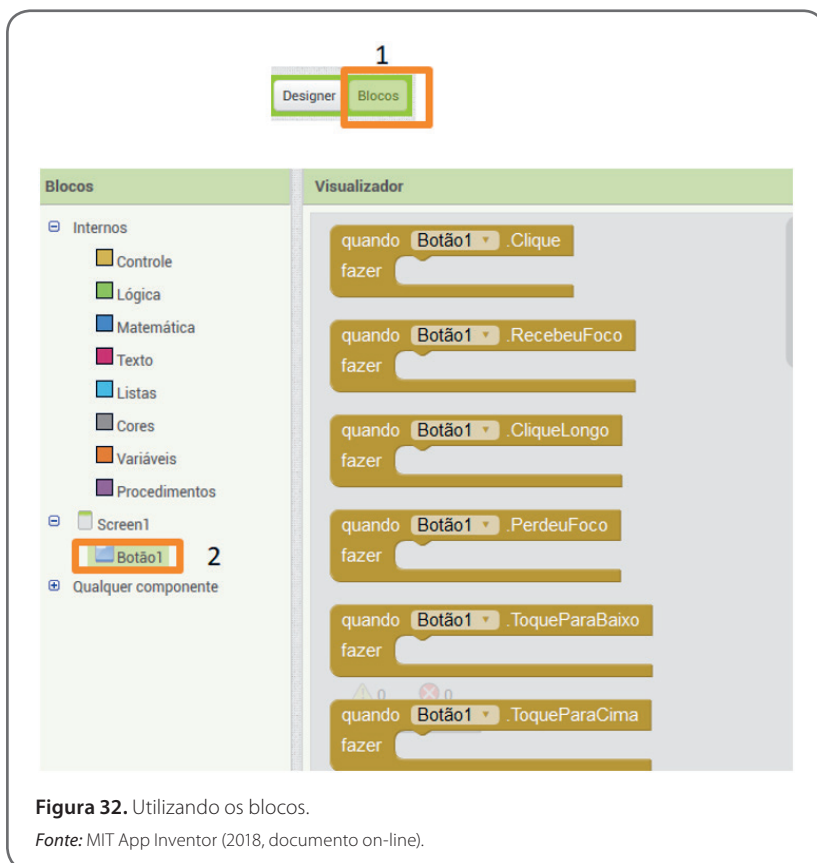
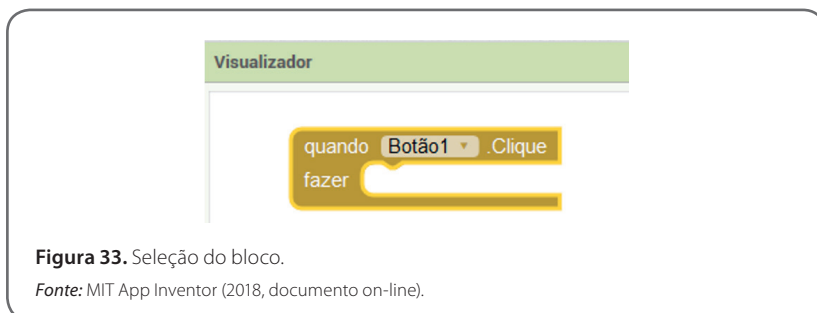
Fonte: MIT App Inventor (2018, documento on-line).

14. Podemos notar, na Figura 32, que diversas opções ficam aparentes para que possamos utilizá-las. Porém, para esse exemplo, iremos selecionar o primeiro bloco. Após a seleção, arraste para o Visualizador, como mostra a Figura 33.

Figura 33. Seleção do bloco.

Fonte: MIT App Inventor (2018, documento on-line).

15. Posteriormente, selecione o componente "Textoparafalar1", e surgirão algumas opções de instruções, como sempre, em formato de blocos. Você irá selecionar a opção que está em destaque na Figura 34.

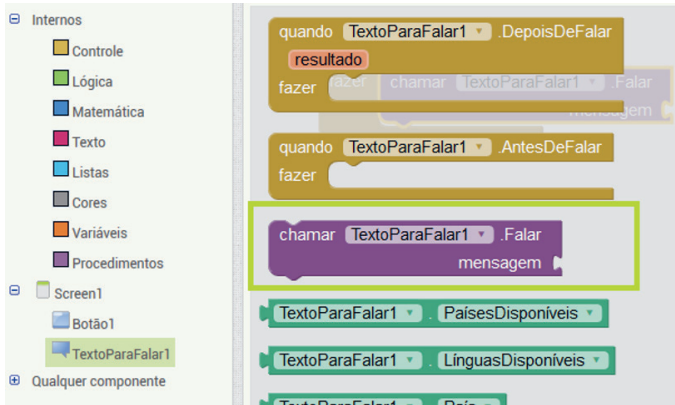



Figura 34. Componente "Textoparafalar".
Fonte: MIT App Inventor (2018, documento on-line).

16. Depois, você vai inserir um componente do tipo "Texto" (Figura 35) e, ao selecioná-lo, surgirão, assim como os demais, diversas opções; selecione a primeira, já que o objetivo dele é traduzir justamente o texto em fala.

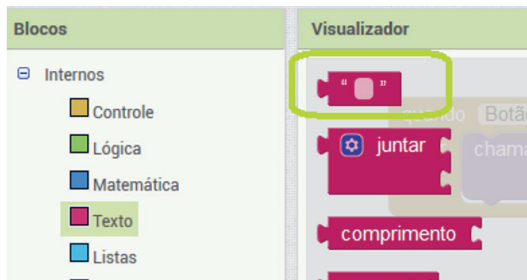



Figura 35. Uso do componente texto.
Fonte: MIT App Inventor (2018, documento on-line).

17. Agora que inserimos alguns componentes, vamos começar a programá-lo. Para isso, você encaixará os blocos que havia selecionado e determinará quais comportamentos estarão dentro do aplicativo. Nesse caso, você está inserindo uma frase, “Meu primeiro aplicativo”, para que possa convertê-la em fala (Figura 36).

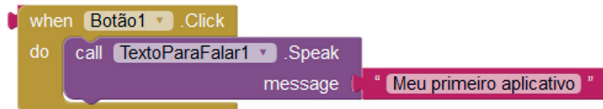



Figura 36. Uso dos blocos.

Fonte: MIT App Inventor (2018, documento on-line).

18. Aproveite e emule seu primeiro aplicativo no emulador. Se você observar, na tela, aparecerá o botão que você criou; para acioná-lo, basta clicar nele e, como você estabeleceu essas ações, ele irá “falar” o texto que você escreveu (Figura 37).

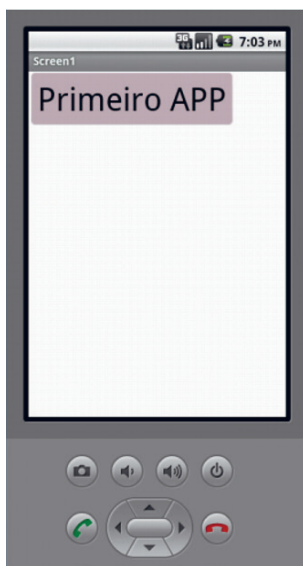



Figura 37. Uso dos blocos.

Fonte: MIT App Inventor (2018, documento on-line).

E aí, qual foi a sensação de acompanhar a execução de seu primeiro aplicativo? Claro que ainda estamos no começo de nossa jornada de estudos sobre o App Inventor, mas já deve ser muito gratificante poder ver a evolução e o funcionamento de um *software* produzido por você.

Você notou que o áudio fica um pouco confuso? Como se fosse uma pessoa estrangeira tentando falar português? Calma, isso tem solução. O componente “TextoparaFalar” (Figura 38) possui algumas propriedades; para isso, volte para o modo de interface Designer para que possa visualizá-las.

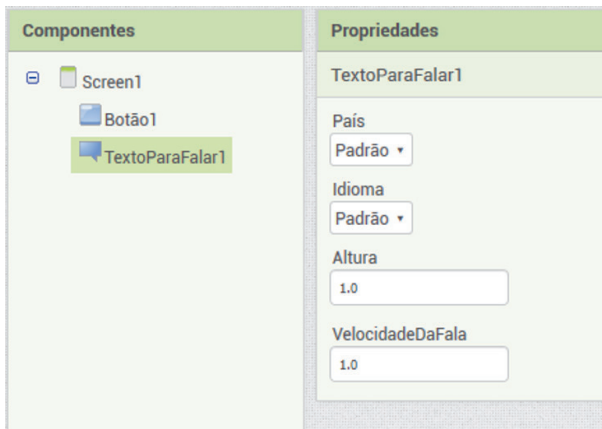



Figura 38. Propriedades do Textoparafalar.
Fonte: MIT App Inventor (2018, documento on-line).

Nessas opções, você pode alterar o país, o idioma, a altura e a velocidade da fala. Aproveite e teste diversas possibilidades — só assim você conseguirá definir o que mais atende aos seus objetivos.

Link

Todas as informações referentes ao MIT App Inventor podem ser encontradas no site do próprio aplicativo:

https://goo.gl/I6TSz

As novidades sobre o App são postadas em diversos blogs, que podem ser acessados por meio do link a seguir.

https://goo.gl/1tmfvL

Outro link recomendado é sobre os conceitos relacionados a lógica de programação e suas aplicações utilizando o App Inventor. Acesse-o a seguir.

https://goo.gl/NCpaHO

Referência

MIT APP INVENTOR. *Anyone Can Build Apps That Impact the World*. 2018. Disponível em: <http://appinventor.mit.edu/explore/>. Acesso em: 06 dez. 2018.

Leituras recomendadas

GERBELLI, N.F.; GERBELLI, V.H.P. *App Inventor*: seus primeiros aplicativos Android. São Paulo: Casa do código, 2017.

MIT APP INVENTOR. *What's New About App Inventor 2?* 2018. Disponível em: <http://appinventor.mit.edu/explore/ai2/whats-new.html>. Acesso em: 06 dez. 2018.

SHERMAN, M.; WALTER, D. *Learning MIT App Inventor:* A hands-on guide to building your own android apps. Upper Saddle River: Addison Wesley, 2014.

SOUZA, A.C.; GÓMEZ, L.A. *Criando Aplicativos no Mit App Inventor.* São Paulo: Visual Books, 2014.

WOLBER, D. et al. *App Inventor 2:* Creat your own android apps. San Francisco: O'Reilly Media, 2014.

DICA DO PROFESSOR

O App Inventor Designer permite a construção da interface do aplicativo. É nesse ambiente que acrescentamos os componentes e definimos suas propriedades, dentre outras funcionalidades.

Confira, na Dica do Professor a seguir, alguns dos recursos do App Inventor.

Conteúdo interativo disponível na plataforma de ensino!

EXERCÍCIOS

1) **O App Inventor é utilizado em diversos projetos direcionados ao sistema operacional Android. Assinale a alternativa que contextualiza de forma correta o App Inventor.**

A) É uma ferramenta paga, utilizada com o objetivo de desenvolver aplicativos para o sistema iOS.

B) É uma ferramenta complexa que possibilita o desenvolvimento de aplicativos para os sistemas operacionais Android e iOS.

C) É uma ferramenta utilizada para desenvolvimento de aplicativos para o sistema iOs e Windows Phone.

D) É uma ferramenta simples, utilizada para desenvolvimento de aplicativos e apenas desenvolvedores experientes conseguem utilizá-la para o sistema operacional Windows Phone.

E) É uma ferramenta lúdica utilizada para desenvolvimento de aplicativos para o sistema operacional Android e que pode ser acessada por meio do próprio *browser*.

2) Assinale a alternativa que apresenta os dois ambientes disponibilizados para desenvolvimento pela ferramenta App Inventor 2.

A) *Designer* e de blocos.

B) Paleta e propriedades.

C) Propriedades e *designer*.

D) Blocos e componentes.

E) Paleta e blocos.

3) Os componentes estão listados em uma seção, a qual mostra lista de botões, campos de texto, mídias, dentre outros. Assinale a alternativa que apresenta o nome dessa seção.

A) Componentes

B) Paleta

C) Multimédia

D) Propriedades

E) Visualizador

4) Assinale a alternativa que apresente a seção na qual os componentes serão inseridos com o intuito de compor o *layout* do aplicativo.

A) Componentes

B) Paleta

C) Visualizador

D) Multimédia

E) Propriedades

5) Os blocos são uma das características mais interessantes do App Inventor, tendo em vista que são eles que deixam a ferramenta mais lúdica. Assinale a alternativa que apresenta a análise de condições e os blocos de instruções a serem executados caso as análises precisem ser realizadas.

A) Controle

B) Lógica

C) Matemática

D) Listas

E) Variáveis

NA PRÁTICA

O mais interessante em desenvolvermos algum *software*, seja ele um aplicativo para dispositivo móvel ou não, é o de podermos testá-los e até fazermos uso dele posteriormente. O app Inventor 2 nos proporciona algumas formas para podermos fazer a emulação e o uso dos aplicativos que criamos na ferramenta.

Confira, Na Prática, como instalar em nosso dispositivo o aplicativo que criamos na ferramenta e, assim, testá-lo.

Conteúdo interativo disponível na plataforma de ensino!

SAIBA MAIS

Para ampliar o seu conhecimento a respeito desse assunto, veja abaixo as sugestões do professor:

App Inventor

Atualmente, podemos ter acesso a diversos recursos, via *web* e o melhor é que muitos deles são gratuitos. Essas ferramentas acabam gerando curiosidade nos desenvolvedores e interessados pela tecnologia. Dessa forma, veremos aqui, uma breve introdução sobre o App Inventor, ferramenta utilizada para desenvolvimento de aplicativos, mantida pelo MIT.

Conteúdo interativo disponível na plataforma de ensino!

Criando projetos

O App Inventor traz, assim como as demais ferramentas de desenvolvimento de *software*, diversos recursos, os quais facilitam o processo de desenvolvimento, pois muitas vezes trazem funcionalidades já prontas para uso e o desenvolvedor apenas realiza configurações para que fiquem de acordo com suas necessidades. Veremos, aqui, conceitos sobre os componentes e suas respectivas propriedades.

Conteúdo interativo disponível na plataforma de ensino!

Configurando o emulador

Apesar de utilizar um ambiente *web* para desenvolvimento, o App Inventor possibilita que possamos desenvolver aplicativos para o sistema operacional Android, e além disso, podemos testá-los em nosso próprio dispositivo. Para isso, iremos aprender, por meio deste vídeo, como

configurar o emulador.

Conteúdo interativo disponível na plataforma de ensino!